Collaborative Information Management using Distributed Multi-Agent System

Rajini Kiran Mai
Asst.Professor, Department of CSE
Brindavan Institute of Technology & Science, Kurnool, Andhra Pradesh

S. Roohinaz
Asst.Professor, Department of CSE
Brindavan Institute of Technology and Science, Kurnool, Andhra Pradesh (of

Shaik Abdul Yunus Basha
Asst.Professor, Department of CSE
Brindavan Institute of Technology and Science, Kurnool, Andhra Pradesh

Abstract—**Emerging of most advanced technologies in wireless systems have spread even in small devices with limited communication and processing speed. Such pervasive systems are more visible offers a specific service to build up more complex services. This goal can be achieved in ad-hoc networks were they dynamically discover and share services between them when they are closer. Meanwhile, the use of Multi-Agent Systems in this environments is necessary to adapt their designs so that they can meet their challenges. More flexible and sophisticated tools are needed to provide an environment where distributed users with similar interests support collaboration.**

In this paper, we studied a trusted communication platform for Multi-Agent System(MAS) and DIAMS to access resources in distributed environments and some theoretical aspects proving the importance of such protocols to have a better communication.

Keywords—Wireless Systems, Pervasive Systems, Ad-hoc networks, MAS, DIAMS.

I. INTRODUCTION

Multi-agent systems can handle most complex application for distributed problem solving. In many applications the agents individual and collective behavior depends on the observed data from various distributed sources where data is a non-trivial problem because of its constraints like limited bandwidth, privacy sensitive data, distributed compute nodes etc. Distributed Data Mining is an area which deals with these challenges to analyze data distributed and offers many algorithmic solutions and mining operations to the resource constraints. As MAS is also a distributed systems environment combines with DDM for data intensive applications. The increasing demand to provide massive data sets incorporated and distributed with limited bandwidth. The computational resources available encouraged the development of distributed data mining. They perform data analysis on individual sites and send the results to other sites where it is required. Many number of DDM solutions are available with various techniques like distributed association rules, distributed clustering, Bayesian learning, classification, and compression, among which only a few of them make use of intelligent agents. Any approach to DDM is a major issue of autonomy and privacy when data can be viewed at different levels and at different perspectives. It may show a greater impact on protecting individual data in terms for their privacy. These issues become particularly important in business application scenarios where different companies may want to collaborate with each other without sharing their personal data to third parties. Recent research work on DDM identified that even without centralized control, cooperation among distributed DM processes is allowed.

In this paper, we broadly studied distributed intelligent agents management system used for collaborative information management.

II. BACKGROUND STUDY

Many research activities are trying to automate the search for distributed algorithm at multi-agent systems level. One among such projects is Behavior Space, which is included with the NetLogo distribution. It allows the user to provide certain patterns with varying input parameters automatically, and recording the results for every run. A user can provide many functions to monitor the state information at every run through which the possible outcomes to the user is provided. Models with low complexity, execution time is not a problem but not for the models where the runtime grows exponentially. Another similar project is behavior space which focuses on sampling the parameter space densely in order to precisely investigate the influence of the different parameters. For a given agent based model it make use of genetic algorithms to search for optimum parameters and doesn't change the structure of the model.

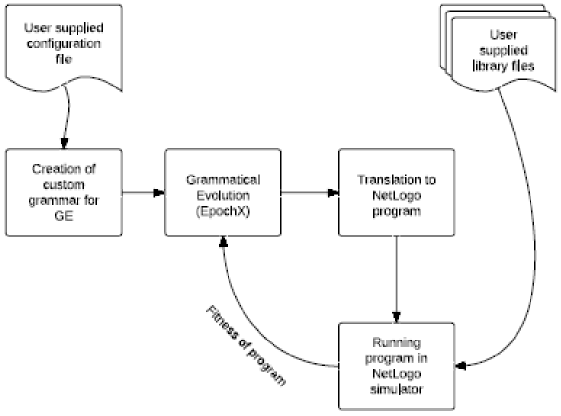
User supplied configuration file
User supplied library files
Creation of custom grammar for GE
Grammatical Evolution (EpochX)
Translation to NetLogo program
Running program in NetLogo simulator
Fitness of program

Figure 1: Meta Compiler Design

This is not possible where the user already knows agent behavior because the only thing left is tweaking of the parameters. Another approach ABM Meta-modeling framework, solves this problem by splitting it into two approaches called bottom-up which is also called forward-

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

mapping problem and top-down also called the reverse mapping problem. These approaches solves the top-bottom problem using the forward mapping to approximate a smooth and continuous surface by representing the space of configurations that would satisfy the system-level requirements set out by the user assuming that the user knows the exact structure of the model. Our approach uses Genetic Programming for altering the internal structure of the code for the agents and gives us limited knowledge about the agent's internal code structure and more power for finding promising novel algorithms. Out of many approaches we chose to use the We further outlined our study for using this approach.

A. Agent and Trust in MAS

Agents are generally classified into stationary agents and mobile agents. Their main difference is that stationary agent resides only on a single platform on which agent operates on and mobile agents are able to migrate from one host to another by preserving their data and its state. Based on the propertied agents can be characterized as interactive, flexible and autonomy.

Scalability, flexibility and fault tolerance are the main advantages of the systems that utilize agents also include an ability to assign different tasks to perform. The most common applications of MAS differs from information filtering and network monitoring, protecting systems from monitoring and management, information filtering and gathering, building self healing, high scalable network systems to transportation, logistics etc. MAS systems are mostly used as tools to build multi-agent systems. Nowadays Classical security model based on central and well secured bastion paradigm is no longer sufficient because in new distributed network agents are ideal attack targets for any malicious operations. Spamming, DoS and spoofing are the most important attacks that can be performed using MAS. On the other hand mobile agents create dynamic environment and establishes trust relations on ad-hoc basis to perform intended tasks more effectively. The major challenging goals are authentication process and authorization decisions where a policy should reflect while changing structure.

Reputation based is a concept of the trust establishment which first utilizes information aggregated by system entities to evaluate reputation of chosen entity and decisions are made according to recommendations from other entities The second credential based TM solution utilizes secure statements and well known credential based platforms like Public Key Infrastructure. Credential based decisions are more reliable and moreover requires well defined semantics.

B. Steganography

Information hiding techniques like network, audio, image and text steganography can became most powerful tool to establish secure and stealth communication among trusted agents. Most of the widely available implementation systems are dedicated to the multimedia applications hiding data in sound files, images and movies. Steganographic solutions are not relatively widespread, but they exist most of them rely on usage of communication protocols. We proposed a distributed steganographic router which will provide ability to create the covert channels between agents. Paths between agents can be built with the use of any of the Steganographic methods in any OSI RM layer and be adjusted to the heterogeneous characteristics of a given network. The concept of a Steganographic router seems to be very accurate to implement such router in this environment and forms a base to develop safe agent communication platform which is required to enhance routing process. System Mixnet proposed there has become a foundation of modern anonymity systems. The concept of Mixnet chaining with encryption has been used in a wide range of applications such as E-mail, Web browsing, and general IP traffic anonymization. Other solutions seem to play a less important role or, as Crowds, can be considered as simplifications of Mixnet. By means of forwarding traffic for others it is possible to provide agents' untraceability. The origin of collaboration intent in this manner can be hidden from untrusted agents and eavesdroppers.

C. Routing in Trust MAS

Routing protocols in IP networks are changing, as the networks evolved, from distance-vector Routing Information Protocol, Interior Gateway Routing Protocol, link-state, Open Shortest Path First and hybrid Enhanced Interior Gateway Routing Protocol protocols for wired networks to proactive Destination-Sequenced Distance Vector, Wireless Routing Protocol, Global State Routing, Optimized Link State Routing, reactive Ad hoc On-demand Distance Vector, Dynamic Source Routing, Light-weight Mobile Routing and hybrid Zone Routing Protocol, Scalable Location Update Routing Protocol, Distributed Dynamic Routing protocols for MANETs. In TrustMAS the most important component that proposed distributed Steganographic router must possess is routing protocol. The effective routing protocol is vital for agents' communication and their performance. The routing protocol for TrustMAS must consider all specific features that are not to find in any other routing environment. That includes anonymity providing random walk algorithm and steganographic methods. Those aspects affect performance of the routing convergence. The first one influences updates due to provide anonymity service they must be periodic. The second one affects links' available bandwidth. That is why the routing protocol for TrustMAS will be designed from the scratch, will be kept as simple as possible soon of the existing routing protocols for MANETs are applicable.

III. DIAMS FOR INFORMATION MANAGEMENT

DIAMS is a distributed intelligent agents system designed to collaborate information management to share and exchange resources on the World Wide Web. This system is designed to help web surfers find needed information from various collection of URLs, as well as from other remote resources. As a result of retrieval process, the system will find a minimal set of most relevant information to the users based on their queries. Relative recall and precision are the two standard measures of identifying effectiveness in traditional information retrieval. Here relative recall is the proportion of relevant information retrieved, whereas precision is the proportion of retrieved information that is relevant. With the abundance of information available on the WWW, it has

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

become more important to have better information access tools that can provide good results in both recall and precision measures. In most of the situation while practicing, unfortunately, it became very difficult to achieve both high recall and precision at the same time.

To address this problem, DIAMS focuses on information accessing and sharing in a distributed environment of information repositories, controlled and managed directly by users. DIAMS does not intend to provide WWW users with information stored in huge and complex public repositories or portals like Yahoo or Lycos. Instead, DIAMS information agents are designed to provide efficient tools to manage and share high quality, well-organized local information repositories customized for users individual needs. DIAMS agents provide complementary information services to that of existing resources available on the WWW. DIAMS provides more other services for stand-alone local information repositories. It is designed to facilitate users with similar interest collaborative information management, sharing and learning among distributed repositories. Information is constantly changing along with user needs and no single user can always maintain the most updated information URLs. Even portals cannot maintain the best information organization to fit all different users with different requirements at different time. Systems of communicating information agents exchange keywords and URLs, but do not communicate with structured information and knowledge. In order to support collaborative information management, DIAMS provides utilities to help users learn about and make use of other users' information.

DIAMS also provides useful information to other users to facilitate information exchange. For users access, DIAMS incorporates a multi-agent architecture organizes and share information on the web. Among several different types of information agents employed, personal agents are the ones that work most directly with users to help support the presentation, organization and management of user information collections. A DIAMS personal agent helps its owner manage their information repository with dynamic organization adaptable to current needs. Flexible hierarchical display is integrated with indexed query search to ensure effective information access. Contents of a repository are kept in object oriented storage to facilitate information sharing. Communication between agents is supported with automatic indexing methods in information retrieval.

IV. MAS SERVICES AND ITS DISCOVERY

Agent technology places a vital role in pervasive computing environments with important challenges. One among them is to allow agents in different devices in an ad-hoc network to share services between. In this paper, we will study a system agent, the Service Discovery Agent that helps other agents in searching services offered by other agents or systems in the network. We identified that none of the service discovery protocols proposed adapts well to the case of pervasive systems, and raised a proposal for a new service discovery protocol, called PDP. The FIPA standard classifies these services into:

1. The Agent Management System manages the life cycle of the agents, the local resources of the platform and the communication channels and provides a "white-pages" service that allow agents to locate each other by their names.
2. The Directory Facilitator is a "yellow-pages" service which identifies what service is providing by which agent.
3. On similar and different platforms, the Agent Communication Channel manages the interchange of messages between agents.

Dynamic service discovery is not a new problem and even there are several proposed solutions proposed for fixed networks with different levels of acceptance. Some of them are SLP, Jini, Salutation and UPnP's SSDP. The Service Location Protocol is an Internet Engineering Task Force standard for enabling IP network-based applications to discover the location of a required service automatically. The SLP defines three "agents" such as User Agents, to perform service discovery on behalf of client software, Service Agents to advertise the location and attributes on behalf of services, and Directory Agents, to store information about the services in the network. SLP has two different modes of operation where a DA is present and collects all service information. Jini technology developed by Sun Microsystems goal is to enable truly distributed computing by representing hardware and software as object oriented objects to form themselves into communities and allows objects to access services on a network in a flexible manner. Jini Service discovery technology is based on a directory service which is similar to the Directory Agent in SLP necessary to the functioning of Jini, and clients to discover services using it, and never can do so directly.

Salutation architecture is for looking up, discovering, and accessing services and information and with a goal to solve the service discovery and utilization problems among a set of applications and equipment in an environment of widespread mobility and connectivity. The Salutation architecture defines Salutation Manager entity that functions as a directory of applications, services and devices, and is generically called Networked Entities. The SLM allows networked entities to discover and use the capabilities of other networked entities Simple Service Discovery Protocol was created as a lightweight discovery protocol for the Universal Plug-and-Play initiative defines a minimal protocol for multicast-based discovery. SSDP works with or without its central directory service. When a service joins the network, it sends an announcement message first to notify its presence to other devices. This announcement is sent by multicast, so that all other devices will discover it, and the present Service Directory, will record the announcement. Meanwhile, the announcement notice may be sent by unicast directly to the Service Directory. A client may ask the Service Directory to discover a service, or it may send a multicast message asking for request.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

These solutions cannot be directly applied to the scenario because they were designed for more suitable wired networks. We studied two main problems in the solutions enumerated: First, most of them use a central server, to maintain the directory services in the network. Pervasive environments cannot be relied upon any single device permanently in order to act as central server, and further and none of the devices present at any moment may be suitable to behave as the server. Second, the solutions are designed to work without a central server, and without considering the power constraints in wireless networks. They make use of broadcast or multicast transmissions which are costless but are power hungry. Accepting that alternatives, we consider two alternative approaches like the "Push" solution, where a device offers a service sends unsolicited advertisements, and the other devices listen to them selecting the services which are interested. "Pull" solution is where a device requests a service when it needs, and devices offering that service with third devices for future use.

In pervasive computing, it is very important to limit the number of transmissions, in order to reduce battery consumption. Meanwhile it is also important to implement mechanisms to detect the availability and unavailability of services when a device joins or leaves the network. These factors need to be considered when selecting between a push solution or a pull solution. The DEAPspace group of the IBM Research Zurich Lab proposed a solution to this problem of service discovery without using a central server. The DEAPspace Algorithm is a pure push solution, where all devices hold a list of all known services with each device periodically broadcasts its view to its neighbours, which can update their views accordingly. We identified the DEAPspace algorithm has the problem of the "world view" of a device spreads from neighbour to neighbour, arriving where some of those services are not available.

In this paper we studied a new service discovery algorithm, the Pervasive Discovery Protocol (PDP), which merges both pull and push solutions characteristics.

V. Pervasive Discovery Protocol Algorithm

This PDP is intended to solve the problem of enumerating the services available in a local cell with less power short-range wireless network, which are composed of limited transmission power, memory, processing power, etc devices. The classical service discovery protocol use a centralised server to listen broadcast or multicast services announcement available at a known port address, and identifies relevant services as response to requested enquiries.

The key objective of Pervasive Discovery Protocol algorithm is to minimise battery use in all devices which means that the number of necessary transmissions should be reduced to identify services as much as possible. A device announces its services only when other devices request them. Service announcements are transferred to all the devices in the network, to know about the new service simultaneously, without having to query for it.

The PDP has two messages: PDP request, which is used to send service announcements and PDP reply, which is used to answer a PDP request, announcing for available services.

A. *PDP Request*

When an application user needs a service offered by the environment then it requests the service from the SDA where the number of broadcast transmissions can be reduced. If a specific service has been requested, the SDA searches for it in all its caches. If it is not found, it forwards the request for that service. If the requested service is available in the network, the SDA updates its caches by sending a PDP request message through all the available interfaces.

B. *PDP reply*

The SDAs in all devices are continually listening on each interface for all messages of PDP requests and PDP replies. When a PDP reply is received, for a service, then it updates their caches accordingly. When a PDP request for a specific service *S* is received, the SDA checks whether the requested service, *S*, is one of its local services and is stored in the cache, or not. If it is, it generates a random time *t*, inversely proportional to the availability time *T* of the device. So, more time of the device is able to offer the service, the higher the probability of the device answering first. During the interval *t*, the SDA listens to the network. If another reply to PDP request arrives, it aborts, otherwise it sends its PDP reply.

VI. Conclusion

DIAMS is a system of distributed agents used to provide better services for users on World Wide Web to access needed information from local and remote repositories. It incorporates MAS architecture and facilitates information sharing. Collaboration between users is facilitated by automated information exchange. It establishes connections between users with similar interests and also incorporates needed services.

PDP Automatic algorithm for distributed systems is one of the contributions to artificial intelligence and agent-based modeling. It has direct applicability in systems where there is a need to have good scalability characteristics.

Acknowledgment

I want to take this opportunity to thank all the people who helped me during my conference sojourn. I understand that it is rather late to acknowledge their contributions, but as the saying goes, better late than never!

References

[1] Krzysztof Szczypiorski, Igor Margasinski, Wojciech Mazurczyk – "Steganographic Routing in Multi Agent System Environment", International Journal of Electronics and Information Technology, vol.4, p.p. 23-27.

[2] Trilok Nath Pandey, Niranjan Panda and Pravat Kumar Sahu – "Improving performance of distributed data mining (DDM) with multi-agent system", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 2, No 3, March 2012, p.p. 74-83.

[3] Sjors van Berkel, Andrei Pruteanu – "Automatic Discovery of Algorithms for Multi-Agent Systems", *GECCO'12 Companion,* July 2012, p.p.7-11.

[4] Vuda Sreenivasa Rao, S Vidyavathi & G.Ramaswamy – "Distributed Data Mining and Agent Mining Interaction and Integration: A Novel Approach", IJRRAS Vol. 4, September 2010, p.p. 388-399.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

[5] Vuda Sreenivasa Rao – “Multi Agent-Based Distributed Data Mining An Overview”, International Journal of Reviews in Computing, 2010, p.p. 83-93.

[6]Lihui Wang, Weiming Shen, Ajit Pardasani – “Collaborative Conceptual Design – State of the Art and Future Trends”, Computer Aided Design, Vol 34, 2002, p.p. 981-996.

[7] James R. Chen & Shawn R. Wolfe, Stephen D. Wragg, “A Distributed Multi-Agent System for Collaborative Information Management and Sharing”, Vol 4, 2012, p.p. 1-4.

[8] Balanovic M., An Adaptive Web Page Recommendation Service. *Autonomous Agents* (1997). Marina Del Rey, CA.

[9] Cohen, W. W. A web-based information system that reasons with structured collections of text. In *Proceedings of the 2nd International Conference on Autonomous Agents* (1998), p.p. 400-407.

[10] Celeste Campo – “Service Discovery in Pervasive Multi Agent Systems”, *Workshop on Ubiquitous Agents on embedded, wearable, and mobile devices,* 2002, p.p. 1-4.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)